

## N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM  
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT  
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED  
IN THE INTEREST OF MAKING AVAILABLE AS MUCH  
INFORMATION AS POSSIBLE

"Made available under NASA sponsorship  
in the interest of early dissemination of Earth Resources  
Program information and without liability  
for any use made thereof."

80-10027

NASA CR-160351  
ERIM 132400-25-R



Technical Memorandum

# A STUDY OF THE TOLERANCE BLOCK APPROACH TO SPECTRAL STRATIFICATION

WYMAN RICHARDSON

AUGUST 1979

(E80-10027) A STUDY OF THE TOLERANCE BLOCK  
APPROACH TO SPECIAL STRATIFICATION

N80-12529

Technical Report, 15 Dec. 1978 - 15 Jun.  
1979 (Environmental Research Inst. of  
Michigan) 60 p HC A04/MF A01

CSCL 13B G3/43 00027

Unclas

Prepared for

**NATIONAL AERONAUTICS AND SPACE ADMINISTRATION**

Johnson Space Center

Earth Observations Division

Houston, Texas 77058

Contract No. NAS9-15476

Technical Monitor: Mr. I. Dale Browne/SF3

 ENVIRONMENTAL  
**RESEARCH INSTITUTE OF MICHIGAN**  
BOX 8618 • ANN ARBOR • MICHIGAN 48107

## NOTICES

Sponsorship. The work reported herein was conducted by the Environmental Research Institute of Michigan under Contract NAS9-15476 for the National Aeronautics & Space Administration, Johnson Space Center, Houston, Texas 77058. I. Dale Browne was Technical Monitor for NASA. Contracts and grants to the Institute for the support of sponsored research are administered through the Office of Contracts Administration.

Disclaimers. This memorandum was prepared as an account of Government sponsored work. Neither the United States, nor the National Aeronautics & Space Administration (NASA), nor any person acting on behalf of NASA:

- (A) Makes any warranty expressed or implied, with respect to the accuracy, completeness, or usefulness of the information, apparatus, method, or process disclosed in this memorandum may not infringe privately owned rights; or
- (B) Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this memorandum.

As used above, "person acting on behalf of NASA" includes any employee or contractor of NASA, or employee of such contractor, to the extent that such employee or contractor of NASA or employee of such contractor prepares, disseminates, or provides access to any information pursuant to his employment or contract with NASA, or his employment with such contractor.

Availability Notice. Request for copies of this memorandum should be referred to:

National Aeronautics & Space Administration  
Scientific & Technical Information Facility  
P. O. Box 33  
College Park, Maryland 20740

Final Disposition. After this document has served its purpose, it may be destroyed. Please do not return it to the Environmental Research Institute of Michigan.

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. 132400-25-R		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle A Study of the Tolerance Block Approach to Spectral Stratification				5. Report Date August 1979	
				6. Performing Organization Code	
7. Author(s) W. Richardson				8. Performing Organization Report No. 132400-25-R	
9. Performing Organization Name and Address Environmental Research Institute of Michigan P.O. Box 8618 Ann Arbor, Michigan 48107				10. Work Unit No.	
				11. Contract or Grant No. NAS9-15476	
				13. Type of Report and Period Covered Technical Memorandum 15 December 1978 through 15 June 1979	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Johnson Space Center Houston, Texas 77058 Mr. I. Dale Browne/SF3				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract <p>The context of this study is the estimation of crop acreage from Landsat multispectral scanner data. The focus is on methods of improving the unsupervised clustering of spectral means of fields into strata. The purpose of the clustering is to isolate the crop and then, from a small sample of fields, identify the strata and estimate acreage. The exact field structure is not required; the clustering is presently applied to quasi-fields formed by grouping together pixels that are spectrally and spatially homogeneous.</p> <p>Three clustering methods are compared: (1) BCLUST, which uses a spectral distance function to accumulate clusters; (2) blocks-alone which divides spectral space into equally-populated blocks; and (3) block-seeds which uses spectral means of blocks-alone as seeds for accumulating distance-type clusters. The equally-populated blocks are constructed by ordering the fields according to a specified channel, dividing them into equally-populated subgroups, similarly dividing each subgroup according to a second channel, and so on for a prescribed set of generating channels.</p> <p>A performance measure for comparing the algorithms is defined, based on a finite without-replacement sample allocated to clusters as nearly as possible proportional to cluster size. The measure is the ratio of the variance of the stratified estimate to that of the unstratified estimate. The comparison was made on 12 winter wheat LACIE segments in Kansas. BCLUST and block-seeds performed equally well and both significantly outperformed blocks-alone. Their average variance ratio</p>					
17. Key Words Clustering, Unsupervised Clustering, Stratified Sampling, Pattern Recognition, Variance Reduction, Tolerance Blocks				18. Distribution Statement	
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 64 + ix	
				22. Price	

16) Cont.

of about .5 showed imperfect separation of wheat from non-wheat. This result points to the need to explore the achievable crop separability in the spectral/temporal domain and suggests evaluating derived features, rather than data channels, as a means to achieve purer spectral strata.

## PREFACE

This report describes part of a comprehensive and continuing program of research concerned with advancing the state-of-the-art in remote sensing of the environment from aircraft and satellites. The research is being carried out for NASA's Lyndon B. Johnson Space Center (JSC), Houston, Texas, by the Environmental Research Institute of Michigan (ERIM). The basic objective of this multidisciplinary program is to develop remote sensing as a practical tool to provide the planner and decision-maker with extensive information quickly and economically.

Timely information obtained by remote sensing can be important to such people as the farmer, the city planner, the conservationist, and others concerned with problems such as crop yield and disease, urban land studies and development, water pollution, and forest management. The scope of our program includes:

1. Extending the understanding of basic processes.
2. Discovering new applications, developing advanced remote sensing systems, and improving automatic data processing to extract information in a useful form.
3. Assisting in data collection, processing, analysis, and ground truth verification.

The research described in this Technical Memorandum was performed under NASA Contract NAS9-15476 during the period from December 15, 1978, through June 15, 1979. I. Dale Browne/SF3 was the NASA Contract Technical Monitor. The program was directed by Richard R. Legault, Vice President of ERIM and Head of the Infrared and Optics Division, Quentin A. Holmes, Program Manager, and Robert Horvath, Head of the Analysis Department.

The work has benefited from technical discussions with Richard J. Kauth, who derived the original reduction of variance factor that is used as one of the performance measures. I was inspired to explore the tolerance block approach to clustering by the lively interest of Richard C. Cicone, who, in addition, contributed creative ideas and editorial assistance. W. Frank Pont contributed to my understanding of stratification in a finite sampling environment. His memorandum on that subject is included as Appendix B. I gratefully acknowledge the help of these co-workers.

It is obvious, but easily overlooked, that this study owes its existence to the supply of good quality Landsat digitized data from Goddard Space Flight Center and Johnson Space Center. Also essential was the pixel-by-pixel ground truth supplied for many LACIE segments which has made it possible to draw conclusions about the relative performance of clustering methods.

# TABLE OF CONTENTS

	<u>Page</u>
PREFACE. . . . .	iii
TABLE OF CONTENTS. . . . .	v
LIST OF ILLUSTRATIONS. . . . .	vii
LIST OF TABLES . . . . .	ix
1. INTRODUCTION. . . . .	1
2. TWO CLUSTERING ALGORITHMS BASED ON TOLERANCE BLOCKS . . . . .	7
3. MEASURE OF PERFORMANCE: THE FIXED SAMPLE REDUCTION OF VARIANCE FACTOR . . . . .	13
3.1 REDUCTION OF VARIANCE FACTOR (RV). . . . .	13
3.2 RV WITH INTEGER ALLOCATIONS. . . . .	14
3.3 THE FIXED-SAMPLE RV. . . . .	15
4. EXPERIMENTS ON 12 KANSAS SEGMENTS TO EVALUATE THE TOLERANCE BLOCK CLUSTERING ALGORITHMS . . . . .	19
4.1 TESTS TO DETERMINE WHICH CHANNELS TO USE FOR TOLERANCE BLOCKING . . . . .	20
4.1.1 TEST OF THE HELPFULNESS OF THE CHANNELS IN BIOWINDOW 1 . . . . .	21
4.1.2 SEARCH FOR THE BEST PAIR OF CHANNELS FOR TOLERANCE BLOCKING. . . . .	21
4.1.3 SEARCH FOR THE BEST CHANNEL SET INCLUDING THE BEST PAIR . . . . .	25
4.2 OPTIMAL NUMBER OF CLUSTERS FOR THE ALGORITHMS. . . . .	26
4.3 COMPARISON OF THE TOLERANCE BLOCK ALGORITHMS WITH BCLUST . . . . .	28
5. CONCLUSIONS AND RECOMMENDATIONS. . . . .	35
5.1 CONCLUSIONS. . . . .	35
5.2 RECOMMENDATIONS. . . . .	36
APPENDIX A: COMPUTER CODE FOR CONSTRUCTING TOLERANCE BLOCKS . .	39
APPENDIX B: EFFECTS OF RANDOM STRATIFICATION. . . . .	49
REFERENCES . . . . .	57
DISTRIBUTION . . . . .	59

# LIST OF ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1	Histogram of Percent Wheat in Quasi-Field Interiors for 12 Kansas Segments. . . . .	2
2	Histogram of Percent Wheat in Spectral Strata For 12 Kansas Segments . . . . .	2
3	Pixel Distribution for BCLUST Strata, Segment 1165. . .	4
4	First Cut to Create Tolerance Blocks. . . . .	8
5	First and Second Cuts to Create Tolerance Blocks. . . .	8
6	Pixel Distributions for Three Clustering Algorithms, Segment 1165. . . . .	11
7	BCLUST Performance as a Function of the Number of Clusters. . . . .	29
8	Performance of Block Seeds as a Function of the Number of Clusters . . . . .	30
9	Performance of Blocks as Clusters as a Function of the Number of Clusters. . . . .	31
A-1	General Outline of the Algorithm for Constructing Tolerance Blocks. . . . .	40
A-2	Listing of XTRAN Code for Tolerance Block Construction.	41

# LIST OF TABLES

<u>Table</u>		<u>Page</u>
1	Table of Combinations of Channel Divisions for Tolerance Blocks and the Number of Blocks Produced. . . . .	9
2	RV Factors Obtained by Running BCLUST With and Without Channels 1 and 2. . . . .	22
3	Comparison of 64-Division Single-Channel Tolerance Blocks	24
4	Comparison of 2-Channel Tolerance Blockings Constructed From Eight Divisions in Each Channel. . . . .	24
5	Comparison of 96-Block Clusterings Constructed from Channel Combinations Containing 3 and 4 . . . . .	27
6	Channel Divisions and Performance of the Tolerance Block Clustering Algorithms as a Function of the Number of Clusters. . . . .	32
7	Comparison of Three Clustering Algorithms at Their Optimal Number of Clusters. Performance Measure is 100-Sample RV . . . . .	33

## INTRODUCTION

This memorandum describes a study whose purpose is to find improved methods of spectral stratification in the context of Procedure M, a system for estimating the acreage of an agricultural crop, such as wheat, from digitized Landsat data [1]. The development of this procedure was stimulated and supported by the Large Area Crop Inventory Experiment (LACIE).

Procedure M as applied to wheat recognition

1. screens and transforms Landsat pixel data from a LACIE segment;
2. clusters the pixels into field-like groups called "quasi-fields" that are homogeneous spectrally and spatially;
3. clusters the quasi-fields spectrally into strata;
4. labels sample quasi-fields from the strata "wheat" or "non-wheat"; and
5. from these labels, forms a stratified sample estimate of the percent wheat in the segment.

Step 3, the clustering of quasi-fields into strata, is designed to separate wheat from non-wheat strata and thereby achieve a sampling efficiency. By this, we mean that a smaller stratified sample will give the same accuracy as an unstratified sample. Another way of putting it is that when the two samples are the same size, the stratified estimate is more accurate.

The grouping of pixels into quasi-fields has been largely successful. Figure 1 is a histogram of the percent wheat in quasi-field interiors. (The interiors consist of pixels faced on all four sides by pixels from the same quasi-field.) This histogram was compiled over all quasi-fields that have interiors from 12 Kansas segments, three acquisitions each. Most of the quasi-fields have less than 10% or more than 90% wheat. Between 10% and 90% wheat, there is only a small scattering of quasi-fields,

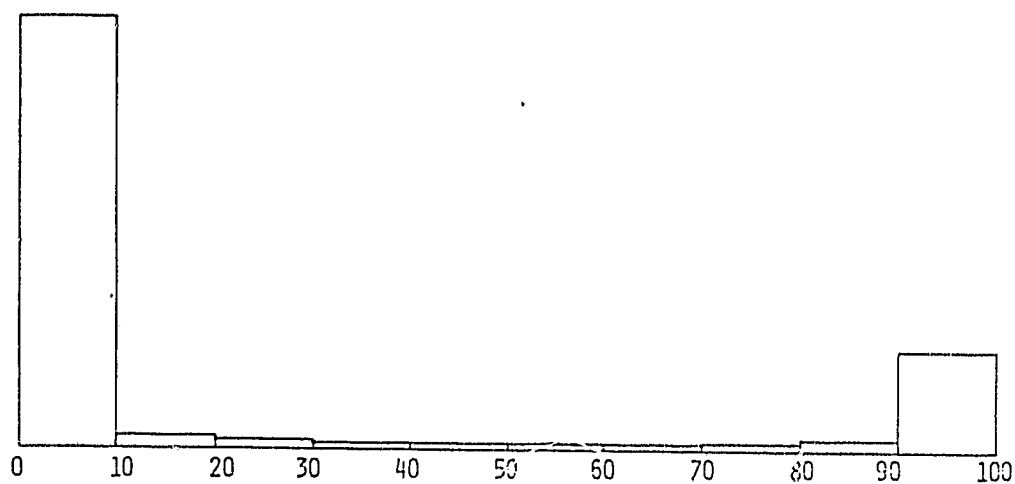


Figure 1. HISTOGRAM OF PERCENT WHEAT IN QUASI-FIELD INTERIORS FOR 12 KANSAS SEGMENTS

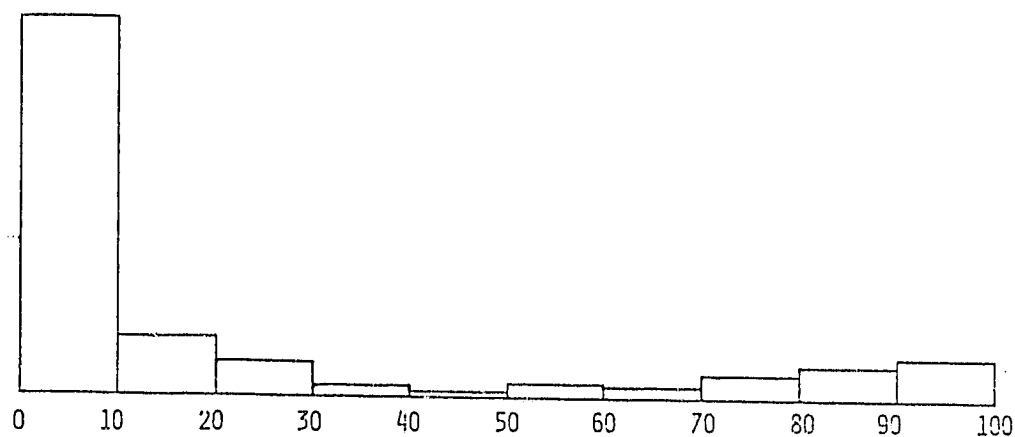


Figure 2. HISTOGRAM OF PERCENT WHEAT IN SPECTRAL STRATA FOR 12 KANSAS SEGMENTS

The picture would not be as pretty if we included edge pixels (i.e., those that are not interior) in the quasi-fields but we would not expect it to be. Edge pixels are often crossed by field boundaries and are the ones that suffer most from misregistration.

The corresponding histogram for strata (Figure 2) shows some mixing of wheat and non-wheat quasi-fields. To make this histogram comparable to the other, the stratum count is weighted by the number of quasi-fields in each stratum. Also for comparability, the histogram is based on quasi-field interior ground truth. So whatever fuzziness is in this histogram is not caused by edge pixels.

A big group of non-wheat quasi-fields are put together into relatively pure strata. The group is not as big as in the quasi-field histogram, for when we compare the two figures, we see that some of the 0 to 10 percent quasi-fields in the quasi-field histogram have spilled over in to the 10 to 20 and 20 to 30 percent bins in the stratum histogram. Similarly, the stack of wheat quasi-fields is spread out into the 80 to 90 and the 70 to 80 bins.

The stratification was carried out by our unsupervised clustering algorithm BCLUST [2]. The question we are considering is whether stratification can be improved by a better clustering algorithm.

One problem with BCLUST is its tendency to produce a few large clusters and many small ones. Figure 3 shows a typical distribution of pixels in a 40-cluster stratification. We try to sample in proportion to the size of the strata because this is the best rule when the stratum wheat proportions are unknown. But in the BCLUST stratification, the big strata are multiply sampled and many small strata are unsampled. Leaving the small strata out would create a bias, so we combine the zero-allocation strata into one wastebasket stratum and sample from it proportional to size. (But we require at least one quasi-field in the sample.) We cannot expect that this wastebasket stratum will be pure, so the sampling from it is inefficient.

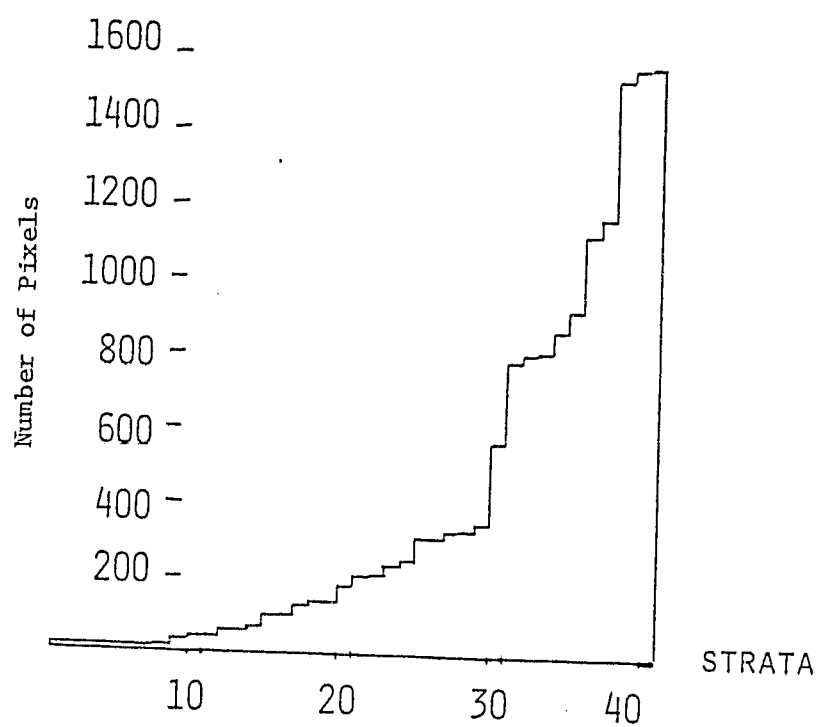


Figure 3. PIXEL DISTRIBUTION FOR BCLUST STRATA  
SEGMENT 1165

The large strata do not have sampling problems if they truly separate wheat from non-wheat. But if they are so large that they mix up the wheat and non-wheat quasi-fields then it would be better to divide them further into smaller strata, more localized spectrally and more homogeneous with respect to crop type.

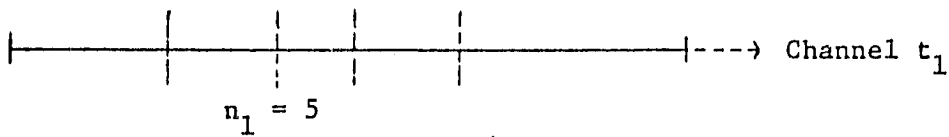
A good clustering algorithm that produced more uniformly-sized strata might improve on the stratification performance of BCLUST. In Section 2, we define two candidate algorithms. In Section 3, we define a performance measure for comparing the three algorithms and in Section 4, describe an experiment to carry out the comparison.

## TWO CLUSTERING ALGORITHMS BASED ON TOLERANCE BLOCKS

An approach to defining a clustering algorithm producing equal-sized clusters is the use of tolerance blocks, an idea suggested to us by R. P. Heydorn [3]. "Tolerance blocks" are equally-populated regions of spectral space constructed as follows. We decide on a small number of channels,  $t_1, \dots, t_k$ , to generate the blocks. We consider the first channel  $t_1$  and order all the quasi-fields according to this channel. We separate this ordered group of quasi-fields into  $n_1$  equal-sized subgroups--equal in the sense of having approximately the same number of pixels (Figure 4). Then we consider each subgroup in turn, order it according to our next channel  $t_2$ , and divide it into  $n_2$  smaller subgroups (Figure 5). We can now consider each one of these smaller subgroups, order it according to our third channel  $t_3$ , and divide it into  $n_3$  still smaller subgroups. We keep this up for all the generating channels specified. The final subgroups are the tolerance blocks,  $n_1, n_2, \dots, n_k$  in all.

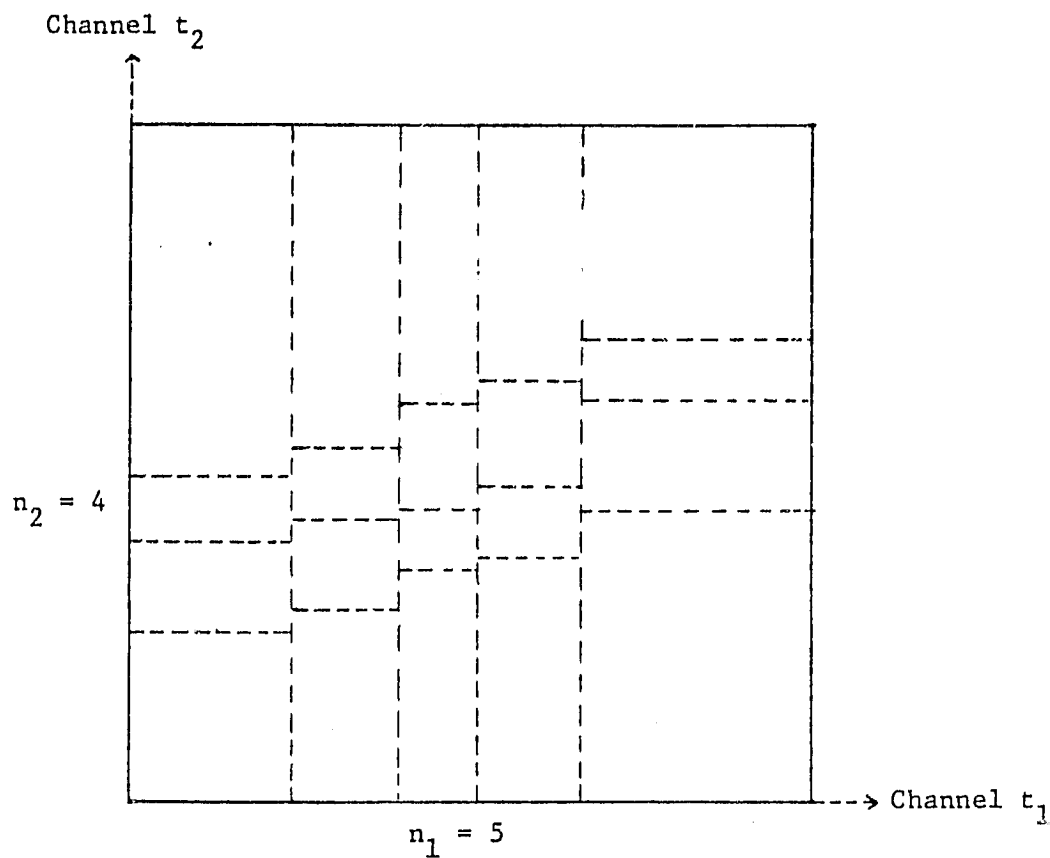
Not all channels need be included in this process. If the same set of channels is used in a different order, the tolerance blocks produced are not necessarily the same. (The results, however, were very similar in our tests.) When channel  $t_2$  is used to divide the first set of subgroups, the points of division will, in general, be different from subgroup to subgroup (column to column in Figure 5). Because we don't cut any quasi-fields in half, but rather assign them to one subgroup or another, the equality of the pixel size of the subgroups can only be approximate.

Table 1 gives a handy reference list of combinations of  $n_1, n_2, \dots, n_k$  and the number of blocks produced for each. A description of computer code for generating tolerance blocks is given in Appendix A.



(The cuts in Channel  $t_1$  separate the quasi-fields into five regions of nearly equal pixel size.)

Figure 4. FIRST CUT TO CREATE TOLERANCE BLOCKS



(The columns are equal-sized groups of quasi-fields separated by cuts in Channel  $t_1$ . The rectangles are equal-sized groups of quasi-fields separated by cuts in Channel  $t_2$ .)

Figure 5. FIRST AND SECOND CUTS TO CREATE TOLERANCE BLOCKS



TABLE 1. TABLE OF COMBINATIONS OF CHANNEL DIVISIONS FOR TOLERANCE BLOCKS AND THE NUMBER OF BLOCKS PRODUCED

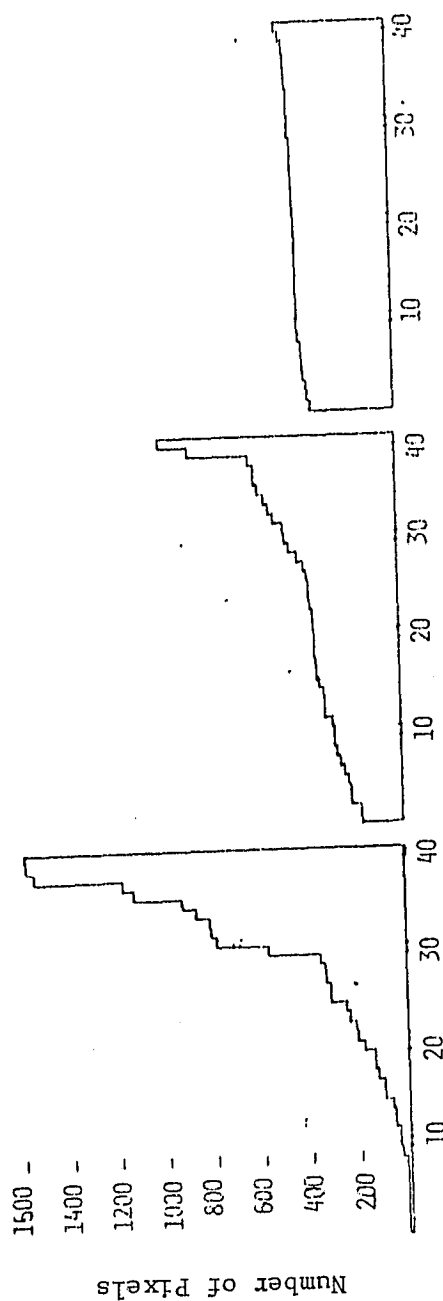
<u>2-Channel</u>			<u>3-Channel</u>			<u>4-Channel</u>							
100:	10	10	100:	4	5	5	108:	3	3	3	4		
99:	9	11	96:	4	4	6	108:	2	3	3	6		
96:	8	12	96:	3	4	8	100:	2	2	5	5		
90:	9	10	90:	3	5	6	96:	2	3	4	4		
88:	8	11	84:	3	4	7	96:	2	2	4	6		
84:	7	12	80:	4	4	5	90:	2	3	3	5		
81:	9	9	80:	2	5	8	84:	2	2	3	7		
80:	8	10	75:	3	5	5	81:	3	3	3	3		
78:	6	13	72:	2	4	9	80:	2	2	4	5		
77:	7	11	72:	3	4	6	72:	2	3	3	4		
72:	8	9	72:	3	3	8	72:	2	2	3	6		
70:	7	10	70:	2	5	7	64:	2	2	4	4		
66:	6	11	64:	4	4	4	60:	2	2	3	5		
64:	8	8	64:	2	4	8	56:	2	2	2	7		
63:	7	9	63:	3	3	7	54:	2	3	3	3		
60:	6	10	60:	3	4	5	48:	2	2	3	4		
56:	7	8	60:	2	5	6	48:	2	2	2	6		
55:	5	11	56:	2	4	7	40:	2	2	2	5		
54:	6	9	54:	3	3	6	36:	2	2	3	3		
50:	5	10	50:	2	5	5	32:	2	2	2	4		
49:	7	7	48:	3	4	4	24:	2	2	2	3		
48:	6	8	48:	2	4	6	16:	2	2	2	2		
45:	6	9	45:	3	3	5							
42:	6	7	42:	2	3	7							
40:	5	8	40:	2	4	5							
36:	6	6	36:	3	3	4	<u>5-Channel</u>						
35:	5	7	36:	2	3	6	108:	2	2	3	3	3	
32:	4	8	32:	2	4	4	96:	2	2	2	3	4	
30:	5	6	30:	2	3	5	96:	2	2	2	2	6	
28:	4	7	30:	2	2	7	80:	2	2	2	2	5	
25:	5	5	28:	2	2	7	72:	2	2	2	3	3	
24:	4	6	27:	3	3	3	64:	2	2	2	2	4	
21:	3	7	24:	2	3	4	48:	2	2	2	2	3	
20:	4	5	24:	2	2	6	32:	2	2	2	2	2	
18:	3	6	20:	2	2	5							
16:	4	4	18:	2	3	3	<u>6-Channel</u>						
15:	3	5	16:	2	2	4	96:	2	2	2	2	2	3
12:	3	4	12:	2	2	3	64:	2	2	2	2	2	2
10:	2	5	8:	2	2	2							
9:	3	3											

9  
8

The blocks are spectrally homogeneous with respect to the generating channels. How homogeneous they are depends on the number of divisions in each channel. But because the number of blocks is the product of the number of divisions, the number of divisions in each channel must be small if we are to end up with a reasonably small number of blocks. So spectral homogeneity of tolerance blocks is limited in two ways: some channels are left out of the block construction and those that are represented may have coarse divisions.

In order to achieve a greater spectral homogeneity, we defined a second tolerance block algorithm that uses all the spectral channels in the clustering process. The tolerance block means are used as seeds distributed like a network throughout spectral space. Around the seeds, clusters are formed by ordinary spectral clustering using a distance function. Although a subset of channels may have been used to create the blocks, all channels are used to compute the block means and carry out the clustering. We hoped to combine in one algorithm the virtues of uniformly-sized clusters and spectral homogeneity.

How well the tolerance block algorithms have succeeded in equalizing the clusters can be seen in Figure 6, a comparison of distributions of strata sizes produced by the three algorithms. BCLUST has a very uneven distribution as we have seen. Many clusters have only a very small number of pixels. When the tolerance blocks themselves are used as clusters, the distribution is very even. When the tolerance blocks are used as seeds, the distribution is less even than for the blocks but considerably more even than for BCLUST.



REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

Figure 6. PIXEL DISTRIBUTIONS FOR THREE CLUSTERING ALGORITHMS  
SEGMENT 1165

MEASURE OF PERFORMANCE: THE FIXED SAMPLE  
REDUCTION OF VARIANCE FACTOR

Although the tolerance block approach to spectral clustering equalizes the size of the strata, the question remains whether it accomplishes its main purpose: to produce strata that discriminate between wheat and non-wheat. To answer this question we developed the measure of stratification performance that is discussed in this section.

### 3.1 REDUCTION OF VARIANCE FACTOR (RV)

The measure of performance heretofore used [4] to evaluate clustering parameters and methods is the reduction of variance factor

$$RV = \frac{\sum_{\text{all strata } i} n_i p_i (1 - p_i)}{np(1 - p)} \quad (1)$$

where  $n_i$  is the number of pixels in stratum  $i$ ,

$p_i$  is the proportion of wheat in stratum  $i$ ,

$n$  is the number of pixels in the segment ( $n = \sum n_i$ ),

$p$  is the proportion of wheat in the segment ( $p = \sum n_i p_i / n$ ).

The RV is the ratio of two variances: the variance of the stratified sample estimate divided by the variance of the unstratified sample estimate. It is a number between 0 and 1. A small number is good. It means that the stratified estimate has a considerably smaller variance than the unstratified estimate and so the stratification is doing some good. We can verify in expression (1) that if the strata are either pure wheat or pure other, then either  $p_i$  or  $1 - p_i$  is 0 and the numerator is 0. If the stratification is worthless, then the  $p_i$ 's are all the same as  $p$  and the factor becomes 1.

### 3.2 RV WITH INTEGER ALLOCATIONS

The RV as a performance measure is unrealistic in two ways. For one thing, it assumes that we are allocating the sample in proportion to the size of the strata. Such an allocation is optimal in the absence of information about the true percent wheat  $p_i$  in each stratum. But it is an approximation because the number of quasi-fields sampled from a stratum must be an integer whereas with few exceptions, the proportional allocation is a real number.

The approximation becomes absurd when the number of strata increases beyond the size of the sample. Then strata must be sampled with a probability rather than with certainty and the variance should rise. But the simple expression (1) does not take account of this effect and continues to decrease (get better) as the number of strata increases.

The approximation is not burdensome when we compare results for clustering algorithms producing approximately equal numbers of strata. But when the numbers are unequal, as when we are trying to find the optimal number of clusters for a given algorithm, the comparison is invalid.

So we can define a better performance measure by assuming a realistic sample size, say 100 quasi-fields, and allocating them to strata as best we can, that is, as nearly as possible proportional to size. If some strata are left unallocated, we'll combine them into a waste-basket stratum and sample it. Then the RV becomes

$$\frac{\sum_{\text{strata } i} \left( \frac{n_i}{n} \right)^2 \frac{p_i(1 - p_i)}{a_i}}{\frac{p(1 - p)}{a}} \quad (2)$$

where  $n_i$  is the number of pixels in stratum  $i$ ,  
 $p_i$  is the proportion of wheat in stratum  $i$ ,  
 $a_i$  is the number of sample quasi-fields allocated to stratum  $i$ ,  
and  $n, p, a$  are the corresponding numbers for the segment.

The allocations  $\{a_i\}$  are made by a subroutine ALLOCB\* as follows:

1. Determine the theoretical allocation  $an_i/n$  for each stratum  $i$ .
2. Round this number to the nearest integer.
3. Collect all the strata with allocation 0 into a wastebasket stratum and allocate sample quasi-fields to it proportional to size, but as least 1. Thus no strata are left out of the sampling.
4. If the integer allocations don't add to  $a$ , multiply the fractional allocations by  $1 + \epsilon$  and repeat.  $\epsilon$  is chosen by an algorithm that makes the procedure rapidly converge. There are, however, some numerical combinations that prevent convergence, and then we settle for an allocation that doesn't quite add up to  $a$ .

The RV with integer allocation (2) is not likely to improve as the number of strata exceeds the sample size because the number of terms being summed in the numerator of (2) remains constant and the wastebasket stratum, in all probability heterogeneous, increases in size.

### 3.3 THE FIXED-SAMPLE RV

A second unrealistic assumption in using expression (1) is sampling with replacement. In fact, it is only reasonable to assume sampling without replacement, implying a hypergeometric, rather than a binomial model.\*\* The effect on the RV is to multiply top and bottom by correction factors as follows:

---

\*ALLOCB is very similar to the allocation subroutine in Procedure M.

\*\*We are indebted to T. Pendleton, Johnson Space Center, NASA, for this suggestion.

$$\text{Fixed-Sample RV} = \frac{\sum \left( \frac{n_i}{n} \right)^2 \frac{p_i(1-p_i)}{a_i} \left( \frac{b_i - a_i}{b_i - 1} \right)}{\frac{p(1-p)}{a} \left( \frac{b-a}{b-1} \right)} \quad (3)$$

where  $n_i$  is the number of pixels in stratum  $i$ ,  
 $p_i$  is the proportion of wheat in stratum  $i$ ,  
 $a_i$  is the number of sample quasi-fields allocated to stratum  $i$ ,  
 $b_i$  is the number of quasi-fields in stratum  $i$ ,  
and  $n, p, a, b$  are the corresponding numbers for the segment.

This is the realistic performance measure that we will use for comparing clustering methods. It is still an approximation because it assumes that all sample quasi-fields are the same size\*.

An implication of the finite collection factors is that stratification incurs a cost. Let us illustrate by an example. Suppose that we create 100 strata, so evenly divided that we allocate one sample quasi-field to each stratum. The correction factor in the numerator is always 1 and drops out. In the denominator,  $b$ , the number of quasi-fields might typically be 400, so the correction factor is  $3/4$ . Now suppose that the stratification completely fails to discriminate, so that  $p_i$  is constantly equal to  $p$ . Then everything cancels out but the  $3/4$  and we are left with a reduction of variance factor of  $1 \frac{1}{3}$ ! This means the variance of the stratified estimate is  $1 \frac{1}{3}$  more than that of the unstratified estimate. Stratification hasn't helped in this case!

This example is extreme because if the stratification were made at random, then just by chance we would expect some  $p_i$ 's to be 0 or 1, and perhaps others to be closer to 0 or 1 than  $p$ . So two opposing forces influence stratification: the finite correction factors penalize stratification and discrimination of wheat from non-wheat rewards it. If the

\* In fact, they are not, and the unbiased scheme used in Procedure M for sampling from unequal-sized quasi-fields [1, pp 31-37] does not have a simply-expressed variance.

stratification is made at random, the two forces would be expected to approximately cancel each other out, as is shown in Appendix B.

A theorem by W. Cochran [5] implies that the simple RV (1) never increases and will usually decrease when any of the strata are broken up into smaller strata. This theorem led us into the comfortable belief that stratification, even if irrelevant, could only help. Cochran's warning that the theorem does not precisely apply to finite sampling is exemplified by our sampling problem, in which the gain or loss from stratification depends on how pure the strata are with respect to the crops of interest. They have to be pure enough to compensate for the finite correction factors or stratification hurts.

# EXPERIMENTS ON 12 KANSAS SEGMENTS TO EVALUATE THE TOLERANCE BLOCK CLUSTERING ALGORITHMS

To evaluate the tolerance block techniques of clustering, we conducted experiments on 1976 LACIE Phase 2 data from 12 segments in Kansas. The application of the data was to measure the amount of winter wheat grown in these segments, so stratum purity was defined as the separation of wheat from non-wheat. The Tasseled Cap transformed channels Brightness and Greenness [2, pp 6-10] from three biowindows were used as follows:

	<u>Brightness</u>	<u>Greenness</u>
Biowindow 1	Channel 1	Channel 2
Biowindow 2	Channel 3	Channel 4
Biowindow 3	Channel 5	Channel 6

The 12 segments were chosen from the blind sites so that ground truth could be used to measure the performance of the clusterings. Only segments with clear data for the three biowindows were used. They were: 1021, 1035, 1165, 1851, 1852, 1861, 1865, 1886, 1163, 1167, 1860 and 1887.

The fixed-sample RV was used as a performance measure with a sample size of 100 assumed. In the remainder of this report, we will multiply the reduction of variance factors by 1000 and refer to the RV (expression 1) and the 100-sample RV (expression 3) as the case may be. The RV will always be, and the 100-sample RV nearly always be, between 0 and 1000, the smaller, the better.

To review, the three algorithms being compared are:

1. BCLUST, which accumulates clusters using a spectral distance function.

2. "Blocks alone", in which the tolerance blocks themselves are clusters.
3. "Block seeds", in which the tolerance block means are used as seeds for accumulating clusters.

Our motivation in this study was to try to improve the clustering so that the strata would achieve purity comparable to that of the blob interiors. Obviously, this goal could not be achieved if we calculated the wheat proportion  $p_i$  from all the pixels in stratum  $i$  whether they be edge or interior. The degradation of the RV factor as we move from quasi-field interiors to the whole quasi-fields would then necessarily be reflected in the RV factor for strata.

The clustering process operates on the means of the quasi-field interiors. What we want to know is whether we can so successfully cluster these interior means that the purity of the clusters (as measured by the RV or the 100-sample RV) approaches that of the quasi-field interiors themselves.

For this purpose, the ground truth of the quasi-field interiors extrapolated to the whole fields is appropriate. Such an extrapolation has been used to provide a close approximation to the percent wheat in a segment. But as we have pointed out, it would result in strata apparently purer than a pixel count would verify. But we aren't interested in purity measured by percent of pixels, but rather purity in the sense that wheat fields are grouped together in strata and so are the non-wheat fields. The kind of purity we are interested in is best measured by the truth that best characterizes the quasi-fields.

#### 4.1 TESTS TO DETERMINE WHICH CHANNELS TO USE FOR TOLERANCE BLOCKING

As discussed previously, we can carry out the tolerance blocking in many different ways (see Table 1). We can use from 1 to 6 channels for the blocking. The fewer channels we use, the more divisions we can

allow in each channel. The order in which the channels are blocked could make a difference. To find a good blocking configuration in a reasonable length of time, we carried out the search in three stages.

#### 4.1.1 TEST OF THE HELPFULNESS OF THE CHANNELS IN BIOWINDOW 1

In the first stage, we conducted a test of the relevance of the data from Biowindow 1. The motivation for the test was that the number of possible combinations of channels is bewildering, and if we could determine that two of the channels were not really helping, we could cut down this number considerably.

The experiment consisted of running BCLUST so that exactly 40 clusters were produced, first using Channels 1...6 and then 3...6. The results are given in Table 2 in terms of the RV. Analogous results would have been obtained with the fixed-sample RV because the finite correction factors would have been similar in each case.

In six of the segments, a substantial reduction in the RV is obtained by including the first two channels. In the other six segments, the difference is trivial. The average difference is 48 points. A t test for differences shows that the significance level of the improvement in the 12 segments is 0.025. There seems to be no relation between the Julian date of pass 1 and the improvement in RV.

We conclude that we cannot dispense with Biowindow 1 in our study of tolerance block clustering.

#### 4.1.2 SEARCH FOR THE BEST PAIR OF CHANNELS FOR TOLERANCE BLOCKING

In the second stage we tested pairs of channels and single channels. The purpose was to find the best pair of channels and include it in a favored position in all the channel combinations tested in the second stage. A second purpose was to compare results from two orderings of the same combination of channels.

TABLE 2. RV FACTORS OBTAINED BY RUNNING BCLUST  
WITH AND WITHOUT CHANNELS 1 AND 2

(The smaller the RV the better.)

<u>Segment</u>	<u>Julian Date of Pass 1</u>	<u>RV Factor</u>	
		<u>With 1&amp;2</u>	<u>Without</u>
1020	92	126	225
1035	312	538	521
1165	326	814	810
1851	19	349	388
1852	295	361	436
1861	349	317	442
1865	349	552	541
1886	311	453	456
1163	70	512	653
1167	70	516	531
1860	294	350	323
1887	311	462	597
Average Difference		-48	
t Value for Difference		2.59	
Significance of t		0.025	

The results of the single-channel test are summarized in terms of RV in Table 3 and of the pair test in Table 4. We would expect analogous results with the fixed-sample RV because of the constant sample size.

The single-channel results identify Channel 4 (greenness in the second biowindow) as the most helpful discriminator of all the channels and indicate that the greenness channels are more helpful than the brightness channels.

The pair results in Table 4 present us with a dilemma: which results are more relevant, those obtained from blocks alone or from block seeds? If we were going to limit the number of channels used in the blocking to two, then the block-seeds results would be most applicable because as we shall see in Section 4.3, the block-seeds RV is consistently lower than the blocks-alone RV.

However, to find the pair of channels that will best combine with other channels to form multi-channel blocks, the blocks-alone results seem most helpful. The seeding operation carries us one computational step away from the effect of separating the data space according to the channel pair. One feels that when the seeding step is applied, differences that showed up in the blocking stage are to some extent averaged out. This conclusion is reinforced by the relative uniformity of the block seeds results in Table 4 and by the invariance of the blocks-alone results over reversed pairs. Therefore, in our search for the best combining pair, we give greater weight to the blocks-alone results. This is why the single-channel test, which was run subsequent to the pair test, has the blocks-alone results only.

Our conclusion is that 3 and 4 (Biowindow 2) are the best combining pair of channels and that, at least in an eight-by-eight blocking, it makes no difference which channel is blocked first. We'll keep our eye on Channel 2 because it showed up well in the single-channel test and was in the only significant pair in the block-seeds pair test.

TABLE 3. COMPARISON OF 64-DIVISION SINGLE-CHANNEL TOLERANCE BLOCKS

		<u>Channel</u>	<u>RV</u>
Biowindow 1	Bright	1	37
	Green	2	-34
Biowindow 2	Bright	3	68
	Green	4	-75*
Biowindow 3	Bright	5	12
	Green	6	-8

(The tabulated number is the average over 12 segments of the difference between the single-channel RV for a segment and the average RV over all single channels for that segment. A negative number is a good score. These results are given for blocks as clusters only.)

\*Difference significant by t test at 0.05 level

TABLE 4. COMPARISON OF 2-CHANNEL TOLERANCE BLOCKINGS CONSTRUCTED FROM EIGHT DIVISIONS IN EACH CHANNEL

<u>Pair</u>	(RV) <u>Blocks as Clusters</u>	(RV) <u>Clusters Seeded by Block Means</u>
3 4	-56*	2
4 3	-54*	1
5 6	14	-7
6 5	10	-10
6 4	4	3
4 6	8	7
5 4	-15	-8
3 6	6	-2
1 4	2	14
2 4	2	-15*
1 2	25	10
2 1	36	-2
1 6	24	0
2 6	-4	6

(The tabulated number is the average over 12 segments of the difference between the pair RV in a segment and the average RV over all pairs in that segment. A negative number is a good score.)

\*Difference significant by t test at 0.05 level

That Channel 3 really does help Channel 4 is shown by the fact that the average RV for Channel 4 alone is 594 and for the pair (3,4) is 489, more than 100 points lower. An interpretation of this difference is that 10% fewer quasi-fields are needed in the sample when Channels 3 and 4 generate the clusters than when Channel 4 alone does.

#### 4.1.3 SEARCH FOR THE BEST CHANNEL SET INCLUDING THE BEST PAIR

The third stage of the channel search was to test various combinations of channels, each combination incorporating the best pair (3,4). Although the two-channel test did not help us decide the order of the channels, other results showed that the last channel in the blocking process is better ordered by the blocking than the earlier channels. So in the tested combinations we put Channel 4 last, Channel 3 next to last and, aside from these two, favored Channel 2.

In all combinations we kept the number of blocks fixed at 96. The patterns of channel divisions were as follows:

<u>Number of Channels</u>	<u>Pattern of Channel Divisions</u>
2	8 12
3	4 4 6
4	2 3 4 4
5	2 2 2 3 4
6	2 2 2 2 2 3

The combinations tested were all possible combinations of the other channels with 3 and 4. The only combinations permuted were (6, 2, 3, 4) and (5, 6, 2, 3, 4). (6, 2, 3, 4) seemed like a good bet because it contains the green channels along with 3 and 4, and (5, 6, 2, 3, 4) seemed also a good five-channel combination to try because it left out Channel 1, which had been indicated to be least effective.

The results of the test are summarized in Table 5. The most remarkable feature of the results is their uniformity. The largest difference from average is 20 points, a modest difference compared with the 75 points that distinguished Channel 4, the 56 points that distinguished the pair (3,4), the 100-point improvement of (3,4) over 4 alone, and the three-figure differences occasioned by leaving out Channels 1 and 2.

When the (3,4) blocking was used as seeds, the RV came out a small, but statistically significant 19 points worse than average. With 8 and 12 divisions in the two channels, it is possible that the seeds were not scattered widely enough in six-dimensional space. Instead of taking 12 divisions in a channel and chopping it up so fine, we might as well take one more channel and divide the three channels into four, four and six parts.

Of the three-channel blockings, (2, 3, 4) seems to be slightly preferable. This (2, 3, 4), the four-channel, five-channel and six-channel combinations are all indistinguishable in performance. We will use a four-channel combination (6, 2, 3, 4) which has the three green channels and the good pair (3,4).

#### 4.2 OPTIMAL NUMBER OF CLUSTERS FOR THE ALGORITHMS

In order to compare the two tolerance block clustering methods with BCLUST clustering, we need to know at what number of clusters, on the average, each algorithm performs best. Then we will have a valid comparison between the algorithms at their best parameter settings.

For each algorithm, we computed the most realistic performance measure, the 100-sample RV, for a variety of numbers of clusters between 16 and 96. A computer program interpolated this number for all integers included in the range and averaged the interpolated value for the 12 segments.

TABLE 5. COMPARISON OF 96-BLOCK CLUSTERINGS CONSTRUCTED  
FROM CHANNEL COMBINATIONS CONTAINING 3 AND 4

<u>Combination</u>	(RV) <u>Blocks as Clusters</u>	(RV) <u>Clusters Seeded by Block Means</u>
3 4	5	19*
5 3 4	11	3
6 3 4	3	11
2 3 4	-4	-5
1 3 4	20	2
5 6 3 4	-9	3
1 2 3 4	4	-6
6 2 3 4	-3	3
2 6 3 4	-5	5
5 2 3 4	-10	-6
1 5 3 4	5	-9
1 6 3 4	3	0
5 6 2 3 4	-3	-11
2 5 6 3 4	-2	-5
1 5 6 3 4	-8	-7
1 6 2 3 4	-5	0
1 5 6 2 3 4	2	2

(The tabulated number is the average over 12 segments of the difference between the combination RV in a segment and the average RV over all combinations in that segment. A negative number is a good score.)

\*Difference significant by t test at 0.05 level

The number of clusters produced by BCLUST is varied by adjusting a parameter  $\tau$ , the greatest distance a quasi-field can be from a cluster mean and still belong to the cluster. Figure 7 shows the graph of BCLUST performance as a function of the number of clusters. It is a smooth curve, because of interpolation and averaging, with a minimum (best score) at about 40 clusters.

The number of clusters produced by the tolerance block algorithms is varied by changing the number of division in the channels that generate the blocks. Table 6 shows the divisions producing 11 cluster numbers between 16 and 96.

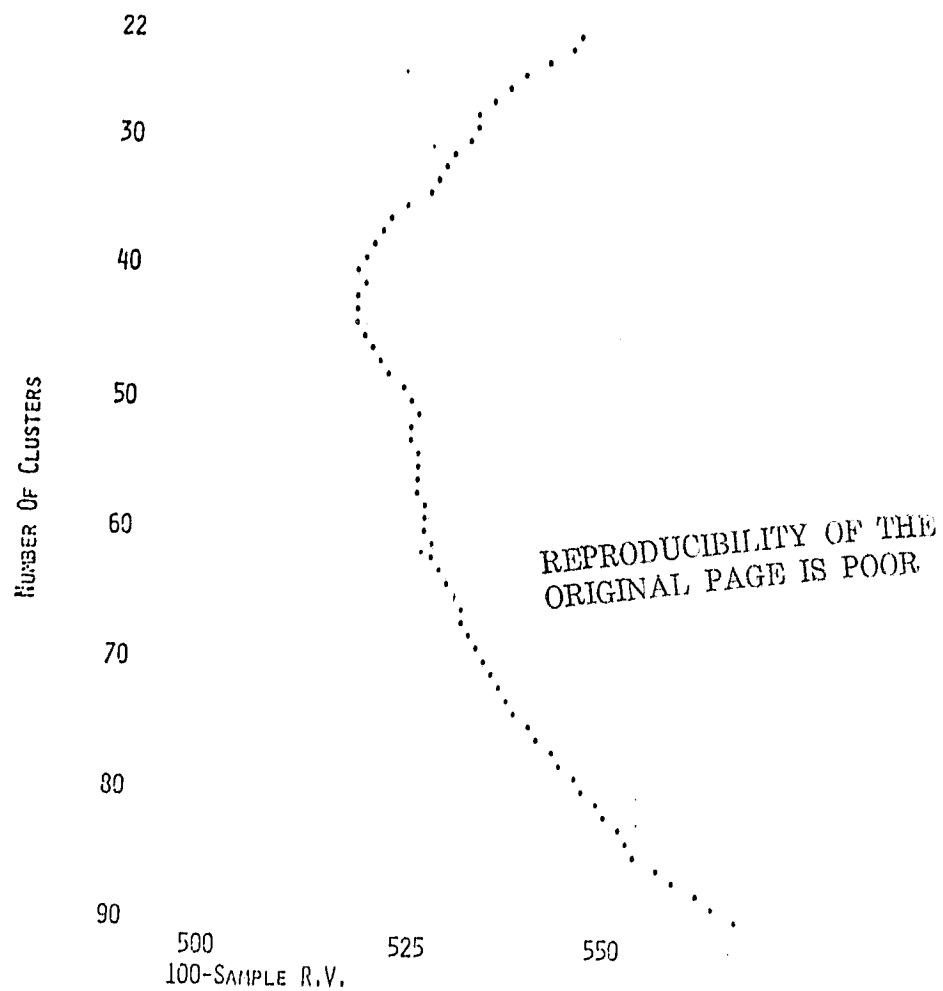
The performance of the tolerance block clustering algorithms as a function of the number of clusters is shown numerically in Table 6 and graphically in Figures 8 and 9. The block-seeds algorithm has a minimum of 40 clusters. The blocks-alone algorithm has a minimum at 48. While the minimum is a razor-thin choice of 48 over 96, the next best numbers are all in the 32 to 54 range, lending support for the validity of a minimum at 48.

In this section we have seen three examples of an optimal number of strata considerably smaller than the sample size--examples of how the benefits from increased stratification were not sufficient to cover the cost of stratification.

#### 4.3 COMPARISON OF THE TOLERANCE BLOCK ALGORITHMS WITH BCLUST

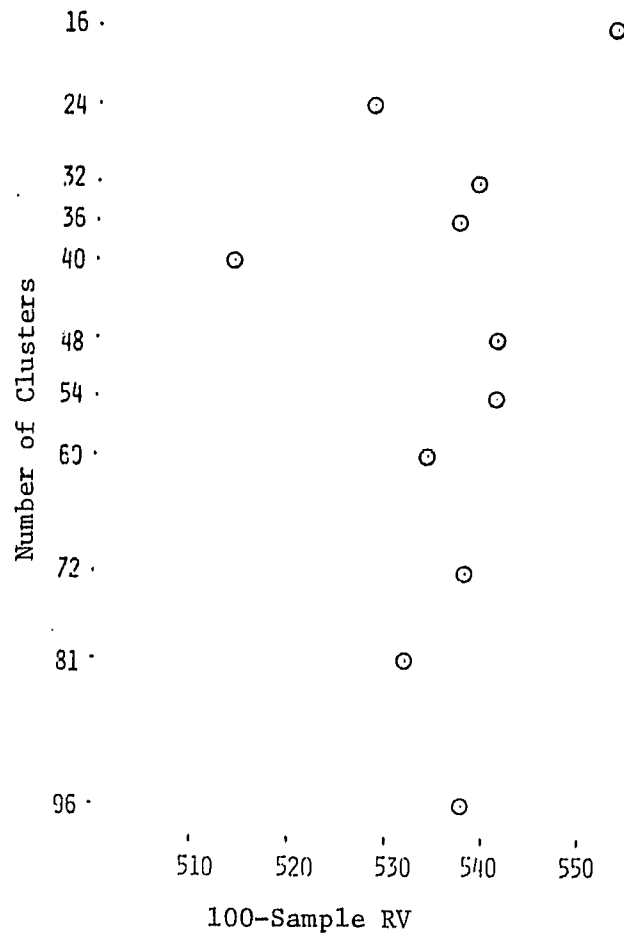
We can now compare the performance of the three clustering algorithms. The performance measure is the 100-sample RV and is measured at the optimal number of clusters for each algorithm: 40 for BCLUST and block-seeds, and 48 for blocks-alone.

The result for each of the 12 Kansas segments and the average results for the 12 is given in Table 7. The blocks-alone algorithm averages 70 points worse than BCLUST and 74 points worse than block-seeds--differences that are significant by a t test. Also, the



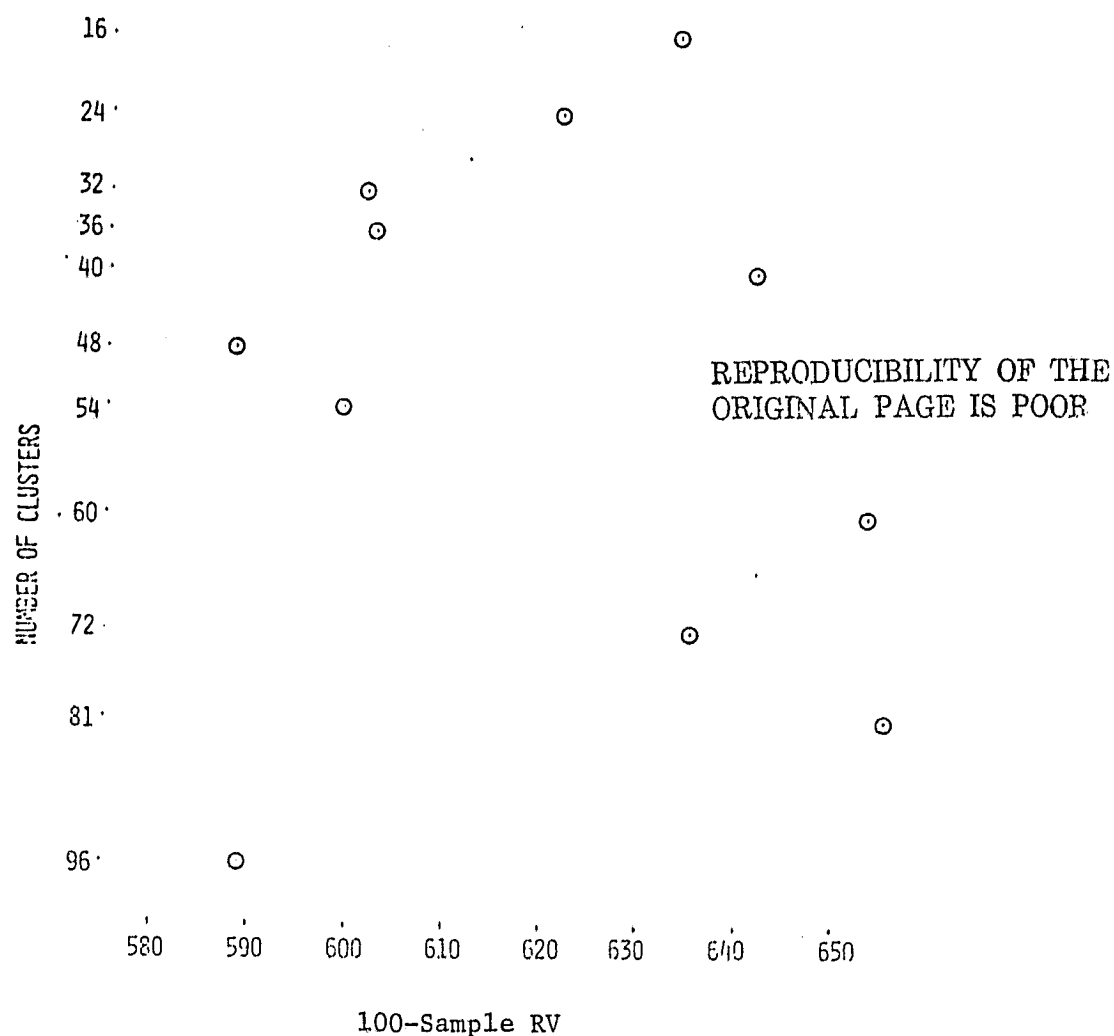
(The performance measure is the 100-sample RV averaged over 12 Kansas segments.)

Figure 7. BCLUST PERFORMANCE AS A FUNCTION OF THE NUMBER OF CLUSTERS



(The performance measure is the 100-sample RV average over 12 segments.)

Figure 8. PERFORMANCE OF BLOCK SEEDS AS A FUNCTION OF THE NUMBER OF CLUSTERS



(The performance measure is the 100-sample RV averaged over 12 segments.)

Figure 9. PERFORMANCE OF BLOCKS AS CLUSTERS AS A FUNCTION OF THE NUMBER OF CLUSTERS

TABLE 6. CHANNEL DIVISIONS AND PERFORMANCE OF THE TOLERANCE  
BLOCK CLUSTERING ALGORITHMS AS A FUNCTION OF THE  
NUMBER OF CLUSTERS

<u>Channels Used For Blocking</u>						
6   2   3   4						
<u>Number of Clusters</u>	<u>Number of Divisions in Each Channel</u>				<u>100-Sample RV</u>	
					<u>Block-Seeds</u>	<u>Blocks-Alone</u>
16	2	2	2	2	553	633
24	2	2	2	3	528	621
32	2	2	2	4	539	601
36	2	2	3	3	537	602
40	2	2	2	5	514	641
48	2	2	3	4	541	588
54	2	3	3	3	541	599
60	2	2	3	5	534	653
72	2	3	3	4	538	635
81	3	3	3	3	532	655
96	2	3	4	4	538	589

(The performance measure is the 100-sample RV averaged over 12 Kansas  
segments.)

TABLE 7. COMPARISON OF THREE CLUSTERING ALGORITHMS AT THEIR OPTIMAL NUMBER OF CLUSTERS PERFORMANCE MEASURE IS 100-SAMPLE RV

Segment	Quasi-Field Interior RV	BCLUST 40 Clusters	Block-Seeds 40 Clusters	Blocks-Alone 48 Clusters
1020	39	181	217	239
1035	187	624	560	578
1165	204	832	922	872
1851	155	383	404	495
1852	136	423	454	614
1861	90	355	389	396
1865	86	610	580	615
1886	168	502	452	532
1163	283	622	621	725
1167	178	652	614	813
1860	145	385	361	420
1887	168	643	588	758
Average	153	518	514	588
Average Difference		4	-74	
t		.31	-3.47	
Significance		Not Significant	Significant at .005	

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

preference is consistent: 11 out of 12 segments for each comparison. Although the clusters produced by blocks-alone have the sampling advantage of uniform size, they are probably less homogeneous spectrally than the clusters from the other algorithms. Two of the channels were not considered at all by blocks-alone, so the clusters would not be very homogeneous in those channels. The channels that were used had 2, 2, 3 and 4 divisions in them, so homogeneity was imperfect. The clusters of the other two algorithms, by contrast, were formed by a spectral distance function and thus emphasized spectral homogeneity.

Between the best tolerance block algorithm (block-seeds) and BCLUST, there is no significant difference. In addition, the preference for one algorithm or the other is equally divided among the 12 segments. Thus, the evidence of this experiment is that tolerance block clustering does not improve spectral stratification.

The "quasi-field interior RV" column, measuring the purity of the interiors of the quasi-fields that make up the strata, is included as a standard of comparison. These low scores show that most of the quasi-field interiors have zero or 100 percent wheat or very close to it. A perfect clustering technique would put the zero percent quasi-fields in some clusters, the 100 percenters in others, and achieve similar RV scores. Yet Table 7 shows a 361-point average difference between the scores. The interior RV was calculated by an expression analogous to (1), so it is not strictly comparable, but even if we raise all the scores in the interior RV column by 1/3 to approximate the effect of the finite sampling correction factors, a tremendous gap remains.

## CONCLUSIONS AND RECOMMENDATIONS

## 5.1 CONCLUSIONS

Two tolerance block techniques and a clustering technique for spectral stratification were evaluated with respect to the estimation of winter wheat acreage in 12 LACIE segments in Kansas. The techniques are (1) to accept tolerance blocks as clusters, (2) to use all-channel means of tolerance blocks as fixed seeds for spectral clustering, and (3) to conduct unsupervised spectral clustering (BCLUST).

Of the two tolerance block techniques, the seeded clustering tested significantly better as measured by the 100-sample reduction of variance factor (a performance measure on the scale of 0 to 1000 that is similar to a previously-defined reduction of variance factor but which, more realistically, takes account of sampling efficiency). Blocks as clusters produced more evenly-sized clusters, which enables efficient sampling, but this advantage was more than balanced by the greater spectral homogeneity of the seeded clusters.

When the tolerance-block-seeded clustering was compared with the unsupervised clustering method BCLUST, there was no significant difference. So in our experiment, the better of the two tolerance block stratification techniques did not show any improvement over previous methods.

A gap of better than 300 points remains between the 100-sample RV scores achieved by our two best stratification methods (about 515) and what is theoretically attainable, the score of 153 for quasi-field interiors.

The optimal number of strata for a sample of size 100 was not found to be 100 or anything close to it, but rather, 40 for BCLUST and the block-seeded algorithm and 48 for the blocks-themselves algorithm.

35

32

The reason the optimum numbers weren't higher is because correction factors applied to finite sampling variances imply a cost to stratification that must be made up by purity of strata. In our experiment, 96 fine strata were not enough purer than 40 coarser strata to defray the cost of the additional strata.

In pursuit of these main conclusions, some subsidiary conclusions were reached.

1. Tolerance block clusters were more uniformly sized than BCLUST clusters, enabling them to be sampled more efficiently. However, this advantage did not result in better stratification performance.
2. Channels in the first biowindow do help the clustering as applied to winter wheat estimation. The reduction of variance score for BCLUST averaged 48 points better when these channels were included.
3. The best channel subsets for generating tolerance blocks contain brightness and greenness from the second biowindow.

## 5.2 RECOMMENDATIONS

The tolerance block study could be carried a little further by investigating the use of tolerance block means as seeds and allowing the updating of means and/or cluster creation and/or iteration of clustering. But the payoff from this effort is likely to be small when we compare the distant goal of relatively pure clusters with the modest scores of the clustering methods tested.

A more promising approach would be to redefine features and test the clustering of these new features using the criterion of the 100-sample reduction of variance factor. The Tasseled Cap features we used in the experiment have the virtue of universal applicability.

Their use implies only that different materials and crops are localized in separate neighborhoods in spectral space. The relative poor performance of the stratification performance indicates the need of features better tailored to the decision problem being considered. Such features could be so specialized that they depend on the crops to be recognized, the confusion crops, the climate, and the prevalent varieties and agricultural practices. There is still room to hope that less special, green-profile-type features [1, pp 20-30] might have a general application to agricultural decision problems.

If better features are found, there could be a greater reward for dividing the feature space into finer strata. Then the sampling advantage gained by the size uniformity of tolerance block clustering could have a greater effect on the performance comparison with BCLUST. So it is too soon to dismiss tolerance block clustering methods from consideration.

The search for features is made in the hope of closing the gap between the RV of .5 found for the strata and the RV of .15 measuring the purity of the quasi-fields. The possible existence of confusion crops inherently inseparable from wheat could define a higher bound than .15 for achievable separability. It may be possible to measure this bound directly, possibly on the basis of a count of identical pairs of data vectors arising from wheat and non-wheat fields, and to chart its value as a function of the acquisitions available. Such a study would give useful feedback in the search for features and provide a warning when multispectral estimation alone is insufficient.

We should not overlook the possibility that other clustering methods might perform significantly better than the ones we tested. CLASSY is now running after much theoretical and practical development. How would it do on the same 12 Kansas blind sites? This would give us another data point for assessing the potential of clustering with our present features and also provide an opportunity to improve the clustering component of Procedure M.

As ways are found to improve spectral clustering, the remaining decision of identifying the clusters becomes less subjective and error prone. In the extreme, we need make only one identification per cluster and this could be done from a smoothed mean value with appropriate attention to historical and economic data. So the finding of better features and clustering methods leads directly to the goal of objective, accurate crop acreage estimation.

APPENDIX A  
COMPUTER CODE FOR CONSTRUCTING TOLERANCE BLOCKS

For convenience, we will use the word "blob" in place of the word "quasi-field" in this appendix.

The general outline of the algorithm for constructing tolerance blocks is given in Figure A.1. Detailed XTRAN code for the construction of tolerance blocks and computation of mean data vectors for tolerance blocks is given in Figure A.2. XTRAN is a language extending FORTRAN in several ways that will be obvious to the reader.

The general outline speaks for itself and we will assume the reader has gone through it. The detailed code contains conventions particular to the clustering program containing it. The following are some notes explaining the code.

We start by assuming that all the blobs being processed are indexed  $L = 1, \dots, QNSS$ . The channel data values in the blobs are contained in a data array  $FDATA(K,L)$  (floating point) or equivalently  $DATA(K,L)$  (integer), where  $K$  is the channel number and  $L$  is the blob index, a number between 1 and  $QNSS$ .

252: Bypass tolerance block construction if  $NTOL$ , the specified number of tolerance blocks, is 0.

255 and 257:  $SEGNO$  is the segment index number of the blob. A group of segments are given indices, say 1-40, for ease of array storage. If  $SEGNO = 0$ , the data point is not a true blob and should be disregarded. If  $IT(SEGNO) = 0$ , the blob is from a segment that the user has decided not to process, so the blob is disregarded.

256:  $PIX(L)$  is the number of pixels in blob  $L$ .

253-261: The  $NBLOB$  acceptable blobs are identified by a  $PO$  vector referring to the index of each acceptable blob.

User specifies channels  $TOL(1) \dots TOL(NTOL)$  for constructing blocks and the number of classes  $NCLASS(1) \dots NCLASS(NTOL)$  each channel divides the data into.

Read blobs  $1, \dots, QNSS$ . The blobs are thus indexed. Some may be unacceptable.

Define the first data group as the NBLOB acceptable blobs.

The group is identified by a position vector  $PO(1) \dots PO(NBLOB)$  giving the index of each acceptable blob.

The algorithm consists of permuting  $PO(1) \dots PO(NBLOB)$  until it orders the blobs into tolerance blocks. Where the blocks begin and end will be shown by a vector  $CL(1) \dots CL(NC)$  giving the number of blobs in each data group. At the end, the data groups are the tolerance blocks. At the start, there is just one data group of all NBLOB blobs. During the algorithm, the data groups are subdivided according to the data values of the channels used for construction.

So to start with,  $CL(1) = NBLOB$  and  $NC = 1$ .

Do the indicated scope for each tolerance channel  $TOL(M)$ ,  $M = 1, \dots, NTOL$ .

```

| Do the indicated scope for each data group I,  $I = 1, \dots, NC$ .
| | Form a vector V of length  $CL(I)$  of channel  $TOL(M)$  values in
| | the data group.
| | Sort V and, at the same time, permute the part of PO corre-
| | sponding to data group I.
| | Cut up data group I into  $NCLASS(M)$  subgroups of nearly equal
| | pixel size, building onto a subgroup vector CC of the
| | numbers of pixels in the subgroups.
| | ↓
| | Make the new NC equal to the total number of subgroups.
| | ↓
| | Move the CC vector to CL.

```

End with  $NC$ : the number of tolerance blocks  
 $CL(1) \dots CL(NC)$ : the number of blobs in each block  
 $PO(1) \dots PO(NBLOB)$ : blob indices ordering the blobs into blocks

Figure A.1. GENERAL OUTLINE OF THE ALGORITHM FOR CONSTRUCTING TOLERANCE BLOCKS

```

248 *
249 * COMPUTE CL(1)...CL(NC), THE SIZES OF THE TOLERANCE BLOCKS
250 * AND A POSITION VECTOR PO ORDERING THE BLOBS BY TOLERANCE BLOCKS
251 * AT PRESENT, THIS OPTION ASSUMES THAT ALL THE PIXELS ARE ON ONE LINE
252 IF NTOL = 0 GO TO ENDTOL
253 NBLN8 = 0
254 DO L=1, QNSS
255   SFGNO = DATA(C19,L)
256   PIX(L) = BASE*DATA(C14,L) + DATA(C15,L)
257   IF SEGNO /= 0 & IT(SEGNO) /= 0
258     NBLN8 = NBLN8 + 1
259     PO(NBLN8) = L
260   END IF
261 END DO
262 NC = 1
263 CL(1) = NBLN8
264 *
265 M = 0; DO WHILE M < NTOL; M = M + 1
266 J = 0
267 NCC = 0
268 TM = TOL(M)
269 NCM = NCLASS(M)
270 *
271 I = 0; DO WHILE I < NC; I = I + 1
272   CLI = CL(I)
273   NPIX = 0
274   DO L=1, CLI
275     V(L) = FDATA(TM, PO(J+L))
276     NPIX = NPIX + PIX(PO(J+L))
277   END DO
278 * SORT V AND AT THE SAME TIME PERMUTE PO(J+1)...PO(J+CLI)
279 CALL VSORTP(V, CLI, PO(J+1))
280 *
281 * SPLIT C(I) UP INTO NCLASS(M) SUBCLASSES OF "EQUAL" PIXEL SIZE
282 LPIX = 0
283 SUMPIX = 0
284 OLDEL = 0
285 NLEFT = NCM
286 QUO = NPIX/NCM
287 L = 0; DO WHILE L < CLI; L = L + 1
288   OLDPPIX = LPIX
289   LPIX = LPIX + PIX(PO(J+L))
290   IF LPIX >= QUO
291     IF LPIX - QUO > QUO - OLDPPIX
292       L = L - 1
293       LPIX = OLDPPIX
294   END IF

```

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

Figure A.2. LISTING OF XTRAN CODE FOR TOLERANCE BLOCK CONSTRUCTION  
(First of Three Pages)

```

295         IF L > OLDL
296             NCC = NCC + 1
297             CC(NCC) = L - OLDL
298             END IF
299             OLDL = L
300             NLEFT = NLEFT - 1
301             SUMPIX = SUMPIX + LPIX
302             IF NLEFT > 0      QUM = (NPIX - SUMPIX)/NLEFT
303             LPIX = 0
304             END IF
305         END WHILE
306     *
307         J = J + CLI
308         END WHILE
309     *
310         CALL MOVER (CC, CL, NCC)
311         NC = NCC
312         END WHILE
313         NCELL = NC
314     *
315     * IF YOU WANT THE TOLERANCE BLOCKS THEMSELVES AS B CLUSTERS:
316     IF BCTOL
317         L = 0
318         DO I=1, NC
319             CLI = CL(I)
320             NO(I) = CL(I)
321             DO J=1, CLI
322                 L = L + 1
323                 PL = PO(L)
324                 DATA(BCHAN,PL) = I
325                 WH = DATA(C21,PL)
326                 S = IT(DATA(C19,PL))
327                 IF WH .EQ. 101
328                     NP(I,S) = NP(I,S) + PIX(PL)
329                     NW(I,S) = NW(I,S) + PIX(PL)*WH
330                 END IF
331             END DO
332         END DO
333         GO TO 7
334     END IF
335 *
336 * TOLERANCE BLOCK DEBUGGING PRINTOUT
337 IF DERUGT
338     WRITE (8, "'0DATA LIST'/")
339     DO L=1, QNSS
340         WRITE (8,106) L, DATA(C14,L), DATA(C15,L), DATA(C21,L),
341             (FDATA(J,L), J=1,NDAT)
342     106     FORMAT (I5, I7, I4, I6, F7.0, 18F5.0)
343     END DO
344     WRITE (8, "'0SORTED DATA LIST'/")
345     END IF
346 *

```

Figure A.2. (continued)

```

347      * COMPUTE TOLERANCE BLOCK MEANS AS SEEDS FOR B CLUSTERING
348      L = 0
349      DO I=1, NC
350          CLI = CL(I)
351          CALL ZERO (4, X(1), X(NDAT) )
352          NPIX = 0
353          WHPER = 0.
354          DO J=1, CLI
355              L = L + 1
356              PL = PU(L)
357              NPIX = NPIX + PIX(PL)
358              DO K=1,NDAT;   X(K) = X(K) + FDATA(K,PL)*PIX(PL);   END
359              IF DEBUGT
360                  WHPER = WHPER + PIX(PL)*DATA(C21,PL)
361                  WRITE (8,107)  PL, PIX(PL), DATA(C21,PL), (FDATA(K,PL),
362                      107      K=1,NDAT)
363                      FORMAT (15, 2I7, 3X, 20F5.0)
364                      END IF
365              END DO
366              FPIX = NPIX
367              CON(I) = 0.
368              DO K=1, NDAT
369                  X(K) = X(K)/FPIX
370                  YK = X(K)*WT(K)
371                  MEAN(K,I) = -2.*YK
372                  CON(I) = CON(I) + YK*YK
373              END DO
374              IF DEBUGT
375                  WHPER = WHPER/FPIX
376                  WRITE (8,108)  I, NPIX, WHPER, (X(K), K=1,NDAT)
377                  108      FORMAT ('0', I4, I7, F7.0, 3X, 20F5.0)
378                  END IF
379              WRITE (8,108A)
380              UPNO(I) = UPTOL
381          END DO
382      ENDTOL:
383      *

```

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

Figure A.2. (continued)

265: Loop through the channels specified for constructing blocks, TOL(M),  $M = 1, \dots, NTOL$ .

266,279,307: J is the index that specifies the part of the PO vector that corresponds to a data group. It starts at 0 and is incremented by CL(I), the number of blobs in data group I.

271: Loop through the data groups  $I = 1, \dots, NC$ .

275: V is built up of the data values of channel TOL(M) from blobs in data group I.

279: VSORTP is a handy subroutine from the International Mathematical and Statistical Libraries that efficiently sorts a vector such as V and, at the same time, permutes another vector, here  $PO(J+L) \dots PO(J+CL(I))$ , the part of the PO vector corresponding to data group I.

281-305: Divide data group I into subgroups of nearly equal pixel size. The output is building onto a long vector, CC, of subgroup sizes, starting with the first data group, and updating NCC, the number of subgroups so far. NCC was set equal to 0 at 267 and is incremented every time a subgroup is defined. NPIX was computed as the number of pixels in data group I (276) and LPIX is the number of pixels currently in the subgroup (289). The idea is to establish a pixel quota QUO, initially NPIX divided by the number NCLASS(M) of subgroups to be established, and keep including blobs in the subgroup until the quota is exceeded (290). At this point, we have to decide whether the current blob L, (or more accurately, the blob identified by  $PO(J+L)$ ) belongs in the current subgroup or the next one. If the number of pixels by which the blob exceeds the quota ( $LPIX - QUO$ ) is greater than the remaining pixels in the blob ( $QUO - OLDPIX$ ), then the blob belongs in the next subgroup. So the blob index L is set back 1 (292) and the number of pixels in the subgroup reverts to the number before the too-big blob was encountered (293).

We don't necessarily update CC and NCC at this point. What if the too-big blob were the first one in the data group? We wouldn't want to count an empty subgroup. So we check the index L of the last blob allowed in the subgroup against OLDL, the index of the last blob in the previously-defined subgroup to avoid this anomaly, if  $L > \text{OLDL}$  (295-298) then the subgroup is non-empty. We define it by incrementing the count NCC of subgroups and appending the number,  $L - \text{OLDL}$ , of blobs in the subgroup to the CC vector.

The new quota QUO is formed by dividing the number of pixels left in the group NPIX-SUMPIX by the number NLEFT of subgroups to be defined. It may be that some groups with one or two large blobs in them cannot be fully divided into NCLASS(M) subgroups.

310: MOVER simply moves CC(1),...,CC(NCC) into the space formerly occupied by the CL vector. It is the new CL.

313: NCELL, used later in the program as the number of clusters, is set equal to the number of tolerance blocks defined.

315-334: The switch BCTOL is set "true" when the tolerance blocks are to be the clusters. Then this section of code is enabled rather than the usual clustering mechanism which is located beyond the tolerance block calculations. This section has to do, therefore, all the chores the clustering mechanism has to perform: the cluster number is included in the data array as the user-specified channel BCHAN (324) and certain running totals are computed to make possible the calculation of the reduction of variance factors (325-330).

340: C14 and C15 are the data channels specifying the number of pixels in Blob L. C21 is the ground truth channel, whose value is an integer between 0 and 101 giving the percent wheat in Blob L. A value of 101 means the ground truth is unknown.

REPRODUCIBILITY OF THE  
ORIGINAL PAGE IS POOR

45  
42

347-381: The means of the data values in all channels are computed for the tolerance blocks, regardless of how many channels were used in the construction of the blocks. This section also contains debugging printout in the sections enabled by the user-set switch DEBUGT.

351: NDAT is the number of data channels containing multispectral data.  $X(1) \dots, X(NDAT)$ , the block mean, is initialized to 0.

348-355: Index I here runs through all the blocks and J runs through all the blobs in each block while L is counting through the blobs as a linear index.

358: The block mean  $\bar{X}$  is a pixel mean computed by weighting the blob channel values by the number of pixels in the blob.

360: WHPER is the percent wheat in the tolerance block. It is computed only for debugging printout. In this statement, WHPER is updated by 100 times the number of wheat pixels in the blob (i.e., the number of pixels in the blob times the percent wheat in the blob). In 375, the cumulated WHPER is then divided by the total number of pixels in the block (357 and 366) to get the wheat percent in the block.

361: In this debugging printout is the original index of the blob, the number of pixels in the blob, the wheat percent in the blob and the mean data vector for the blob (computed as always from the interior pixels).

371,372: The block mean is stored as a cluster seed for later use in the clustering program. The cluster constant  $CON(I)$  and the multiplication of the mean by  $WT(K)$  and  $-2$  are peculiarities of the clustering program. [Instead of computing  $(X - \bar{X}_i)^2$  for the data point  $X$  and the cluster mean  $\bar{X}_i$ , the program multiplies it out and computes  $X^2 - 2\bar{X}_i + \bar{X}_i^2$  for each cluster. Because  $X^2$  is the same for each cluster, it is omitted and the  $i$  minimizing  $-2\bar{X}_i + \bar{X}_i^2$  is chosen.  $CON(I)$  is  $\bar{X}_i^2$ .  $WT(I)$  is to allow for weighted clustering.]

376: In this debugging printout is the block number, the number of pixels in the block, the wheat percent in the block and the mean data vector for the block.

APPENDIX B  
EFFECTS OF RANDOM STRATIFICATION  
(W. Frank Pont)

B.1 INTRODUCTION

It is shown in Section 3 that the stratified random sampling variance could be larger than the simple random sampling variance of the same size when sampling without replacement. It should be pointed out that there are two conflicting factors which affect the sample variance. These are:

1. Grouping the population elements into strata whose proportions of grain are possibly closer to 0 and 1 than the proportion of grain in the population. This factor tends to lower the variance of the stratified proportion estimate compared to the variance of the unstratified proportion estimate.
2. The number of samples which are obtainable in stratified sampling without replacement is smaller than the number of samples obtainable in simple random sample of the same size. This factor tends to increase stratified sampling variance compared to the variance of the unstratified proportion estimate. Sometimes, factor 2 outweighs factor 1.

R. Kauth pointed out that while stratification based on spectral variables could be defined in such a way as to make the proportion of grain in each stratum nearly equal (hence defeat the purpose of stratifying), it is very unlikely that stratification which assigns elements which look alike into the same stratum would have this effect. The worst that should happen in spectral stratification is that the spectral characteristics of the elements might have nothing to do with the true label, in which case the stratification would be random with respect to the true labels. Thus, his conjecture was: The probability structure and sampling variance of random stratification followed by stratified sampling is the

same as simple random sampling of the same sample size. Using a restricted definition of random stratification, we show that the probabilities of obtaining a given sample are the same under both sampling models. However, the variance of simple sampling proportion estimate is smaller unless the stratified sampling is proportional to size.

## B.2 NOTATION AND CONVENTIONS

We assume that a sample of size  $n$  is to be selected from a population of size  $N$ . We assume that the elements of the population,  $b=1,2,3,\dots,N$ , are assigned at random in such a way that there are  $Q$  strata, denoted as  $s=1,2,3,\dots,Q$  of size  $N_1, N_2, N_3, \dots, N_Q$ . In the stratified sampling,  $n_s$  elements are to be selected from stratum  $s$ . We note the two relations:

$$N = N_1 + N_2 + \dots + N_Q$$

and

$$n = n_1 + n_2 + \dots + n_Q$$

A stratification is a function which associates every element  $b$  with some stratum  $s=1,2,\dots,Q$  with the above restrictions. Formally, the function

$$i:\{1, 2, 3, \dots, N\} \mapsto \{1, 2, 3, \dots, Q\}$$

is a stratification if, for  $s=1,2,3,\dots,Q$ , the cardinality of the set

$$\{b:i(b) = s\} = N_s$$

There are  $\binom{N}{N_1 N_2 \dots N_Q}$  possible stratifications. We use  $i$  to denote a fixed stratification and use  $I$  to denote a stratification chosen at random from all possible stratifications. We also view a sample as a function which tells us whether an element  $b$  is in the sample or not. The sampling function is

$$j:\{1, 2, 3, \dots, N\} \mapsto \{0,1\}$$

$j(b)=1$  if  $b$  is in the sample,  
 $=0$  otherwise.

There are  $\binom{N}{n}$  possible simple random samples and

$$\sum_{s=1}^Q \pi \binom{N_s}{n_s}$$

stratified random samples with respect to a fixed stratification  $i$ . We use  $j$  to denote a fixed sample (one which has already been chosen) and  $J$  to denote a sample which is to be randomly selected from all possible samples.

A sample function  $j$  and a stratification function  $i$  are compatible, denoted  $i \sim j$ , if sample  $j$  could have been obtained for the stratification  $i$ . That is, the cardinality of the set  $\{b: i(b) = s \text{ and } j(b) = 1\}$  is  $n_s$  for every  $s=1,2,\dots,Q$ .

We assume every element  $b$  has a label 1 or 0 (grain or non-grain in our case) denoted as  $L(b)$ . Since we are not only choosing the sample  $j$  at random but also the stratification, we need to define the probabilities associated with  $I$ ,  $J$  and  $(I,J)$ .

$$P_I(i) = \binom{N}{N_1 N_2, \dots, N_Q}^{-1}$$

is the probability that  $i$  is chosen as the stratification;

$$P_J(j) = \binom{N}{n}^{-1}$$

is the probability that sample  $j$  is chosen;

$$P_{I|J=j}(i) = \binom{n}{n_1 n_2, \dots, n_Q}^{-1} \binom{N-n}{N_1-n_1 N_2-n_2 \dots N_Q-n_Q}^{-1}$$

if  $i \sim j$ . Otherwise,

$$P_{I|J=j}(i) = 0$$

This is the conditional probability that stratification  $i$  is chosen given that sample  $j$  has been chosen. This result can be obtained by direct counting or by the use of Bayes theorem.

$$P_{J|I=i}(j) = \frac{Q}{\pi} \left( \frac{N_s}{n_s} \right)^{-1}_{s=1}$$

if  $i \neq j$ . Otherwise,

$$P_{J|I=i}(j) = 0$$

is the conditional probability that sample  $j$  is selected given that stratification  $i$  has been chosen.

The joint probability can be defined as:

$$P_{IJ}(i,j) = P_I(i)P_{J|I=i}(j) \text{ or}$$

$$P_{IJ}(i,j) = P_J(j)P_{I|J=j}(i).$$

In the first case, for  $i \neq j$

$$\begin{aligned} P_{IJ}(i,j) &= P_I(i)P_{J|I=i}(j) \\ &= \left( \frac{N}{N_1, N_2, \dots, N_Q} \right)^{-1} \frac{Q}{\pi} \left( \frac{N_s}{n_s} \right)^{-1}_{s=1} \\ &= \frac{\frac{Q}{\pi} \prod_{s=1}^Q N_s!}{N!} \frac{Q}{\pi} \frac{n_s!(N_s - n_s)!}{N_s!}_{s=1} \\ &= \frac{\frac{Q}{\pi} \prod_{s=1}^Q n_s!(N_s - n_s)!}{N!} \end{aligned}$$

In the latter case, for  $i \neq j$

$$\begin{aligned} P_{IJ}(i,j) &= P_J(j)P_{I|J=j}(i) \\ &= \left( \frac{N}{n} \right)^{-1} \left( \frac{n}{n_1, n_2, \dots, n_Q} \right)^{-1} \left( \frac{N-n}{N_1 - n_1, N_2 - n_2, \dots, N_Q - n_Q} \right)^{-1} \\ &= \frac{n!(N-n)!}{N!} \left( \frac{\frac{Q}{\pi} \prod_{s=1}^Q n_s!}{n!} \right) \left( \frac{\frac{Q}{\pi} \prod_{s=1}^Q (N_s - n_s)!}{(N-n)!} \right) \\ &= \frac{\frac{Q}{\pi} \prod_{s=1}^Q n_s!(N_s - n_s)!}{N!} \end{aligned}$$

which is consistent with the first definition. Note in both cases,

$$P_{IJ}(i,j) = 0$$

if  $i \neq j$ .

Now that we have defined the joint probability of I and J, we can view  $P_I$  and  $P_J$  as marginal probabilities, that is

$$P_I(i) = \sum_j P_{IJ}(i,j)$$

and

$$P_J(j) = \sum_i P_{IJ}(i,j).$$

A consequence of this is that the probability of obtaining a fixed sample  $j$  is the same in simple random sampling and in random stratification followed by stratified random sampling.

We now examine the two estimates. The simple random sample proportion  $\bar{P}$  depends only on  $j$ , namely

$$\bar{P}(j) = \frac{\sum_{b:j(b)=1} L(b)}{n}$$

Given a stratification  $i$ , we can introduce  $i$  into this relation

$$\bar{P}(j) = \sum_{s=1}^Q \frac{n_s}{n} \sum_{\substack{b:i(b)=s \\ j(b)=1}} \frac{L(b)}{n_s} = \sum_{s=1}^Q \frac{n_s}{n} \bar{P}_s(i,j) \quad (1)$$

where

$$\bar{P}_s(i,j) = \frac{1}{n_s} \sum_{\substack{b:i(b)=s \\ j(b)=1}} L(b)$$

The stratified random sample proportion  $\hat{P}$  is defined by

$$\hat{P}(i,j) = \sum_{s=1}^Q \frac{N_s}{N} \bar{P}_s(i,j) \quad (2)$$

We note that (1) and (2) are equal if

$$\frac{n_s}{n} = \frac{N_s}{N}$$

where  $s=1,2,3,\dots,Q$ ; i.e., if we are able to sample proportional to size.

### B.3 SAMPLE VARIANCE UNDER THE TWO MODELS

In this section, the letter "E" will stand for "expectation of" and "V" for "variance of".

$$E_{IJ}\hat{P}(I,J) = E_J\bar{P}(J) = E_{J|I=1}\hat{P}(I,J) = P$$

where

$$P = \frac{1}{N} \sum_{b=1}^N L(b)$$

$$E_{I|J=j}\hat{P}(I,J) = E_{I|J=j}\bar{P}_s(I,J) = \bar{P}(j) \quad (3)$$

$$\begin{aligned} V(\hat{P}(I,J)) &= E_{IJ}(\hat{P}(I,J) - P)^2 \\ &= E_{IJ}(\hat{P}(I,J) - \bar{P}(J) + \bar{P}(J) - P)^2 \\ &= E_{IJ}(\hat{P}(I,J) - \bar{P}(J))^2 + 2E_{IJ}((\hat{P}(I,J) - \bar{P}(J))(\bar{P}(J) - P)) \\ &\quad + E_{IJ}(\bar{P}(J) - P)^2 \\ &= E_{IJ}(\hat{P}(I,J) - \bar{P}(J))^2 + 2E_{IJ}((\hat{P}(I,J) - \bar{P}(J))(\bar{P}(J) - P)) \\ &\quad + V(\bar{P}(J)) \end{aligned} \quad (4)$$

$$\begin{aligned} E_{IJ}[(\hat{P}(I,J) - \bar{P}(J))(\bar{P}(J) - P)] \\ &= \sum_{(i,j) \in \mathcal{I} \times \mathcal{J}} [(\hat{P}(i,j) - \bar{P}(j))(\bar{P}(j) - P)] P_{IJ}(i,j) \\ &= \sum_j \left[ \sum_{i: i \in \mathcal{I}_j} (\hat{P}(i,j) - \bar{P}(j)) P_{I|J=j}(i) \right] (\bar{P}(j) - P) P_J(j) \end{aligned} \quad (5)$$

$$\begin{aligned}
 & \sum_{i:i \sim j} (\hat{P}(i,j) - \bar{P}(j)) P_{I|J=j}(i) \\
 &= \sum_{i:i \sim j} \left[ \sum_{s=1}^Q \left( \frac{N_s}{Nn_s} - \frac{1}{n} \right) \sum_{\substack{b:i(b)=s \\ j(b)=1}} L(b) \right] P_{I|J=j}(i) \\
 &= \sum_{s=1}^Q \sum_{i:i \sim j} \left( \frac{N_s}{Nn_s} - \frac{1}{n} \right) \sum_{\substack{b:i(b)=s \\ j(b)=1}} L(b) P_{I|J=j}(i) \\
 &= \sum_{s=1}^Q \sum_{i:i \sim j} \left( \frac{N_s}{N} - \frac{n_s}{n} \right) \bar{P}_s(i,j) P_{I|J=j}(i) \\
 &= \sum_{s=1}^Q \left( \frac{N_s}{N} - \frac{n_s}{n} \right) \sum_{i:i \sim j} \bar{P}_s(i,j) P_{I|J=j}(i) \\
 &= \sum_{s=1}^Q \left( \frac{N_s}{N} - \frac{n_s}{n} \right) \bar{P}(j) \quad \text{by (3)}
 \end{aligned}$$

$$= 0. \quad (6)$$

We now have from (4), (5) and (6)

$$\begin{aligned}
 V(\hat{P}(I,J)) &= E_{IJ} (\hat{P}(I,J) - \bar{P}(J))^2 + V(\bar{P}(J)) \\
 &\geq V(\bar{P}(J))
 \end{aligned}$$

$$V(\hat{P}(I,J)) = V(\bar{P}(J)), \text{ if } \frac{n_s}{n} = \frac{N_s}{N}.$$

We conclude that the variance is increased by stratification which is random with respect to the labels unless the strata sample sizes are proportional to strata size, i.e.,

$$\frac{n_s}{n} = \frac{N_s}{N}$$

## REFERENCES

1. Cicone, R., E. Crist, R. Kauth, P. Lambeck, W. Malila and W. Richardson, Development of Procedure M for Multicrop Inventory, with Tests of a Spring-Wheat Configuration (Final Report), ERIM Report No. 132400-16-F, Environmental Research Institute of Michigan, Ann Arbor, Michigan, March 1979.
2. Kauth, R. J. and W. Richardson, Procedure B: A Multispectral Training Selection and Proportion Estimation Procedure for Processing Landsat Agricultural Data (Final Report), ERIM Report No. 122700-31-F, Environmental Research Institute of Michigan, Ann Arbor, Michigan, November 1977, pp 89-94.
3. Heydorn, R. P., Multisegment Training for the LACIE, Note attached to a letter from R. P. Heydorn, Johnson Space Center, National Aeronautics and Space Administration, to R. F. Nalepka, Environmental Research Institute of Michigan, September 24, 1976.
4. Kauth, R. J. and W. Richardson, Task 1, Multisegment Training, Analysis of Scanner Data for Crop Inventories, Third Quarterly Report on NASA Contract NASA9-15476, ERIM Report No. 132400-12-P, Environmental Research Institute of Michigan, September 1978, pp 1-5.
5. Cochran, W. G., Sampling Techniques, Third Edition, Wiley, 1977, pp 99 ff.